

Tableau Proof Systems for Justification Logics

Meghdad Ghari

School of Mathematics, Institute for Research in Fundamental Sciences (IPM),
P.O.Box: 19395-5746, Tehran, Iran
ghari@ipm.ir

Abstract. In this paper we present tableau proof systems for various justification logics. We show that the tableau systems are sound and complete with respect to Mkrtychev models. In order to prove the completeness of the tableaux, we give a syntactic proof of cut elimination. We also show the subformula property for our tableaux.

Keywords: Justification logics, Tableaux, Subformula property, Analytic cut, Cut elimination

1 Introduction

Justification logics are modal-like logics that provide a framework for reasoning about epistemic justifications (see [3, 4, 11] for a survey). The language of justification logics extends the language of propositional logic by justification terms and expressions of the form $t : A$, with the intended meaning “ t is a justification for A ”. Justification terms are constructed from variables and constants by means of various operations. The first logic in the family of justification logics, *the Logic of Proofs* LP, was introduced by Artemov in [1, 2]. The logic of proofs is a counterpart of modal logic S4. Other logics of this kind have been introduced so far (cf. [13]). In this paper we deal only with those justification logics which are counterparts of normal modal logics between K and S5.

Various tableau proof systems have been developed for the logic of proofs (see [8, 10, 15, 16]). However, it seems the only analytic tableau proof system is Finger’s KE tableaux for the logic of proofs [8]. Finger’s tableau system has KE tableau rules (cf. [7]) in its propositional part. KE tableaux have linear tableau rules for propositional connectives, and the cut rule (*PB*).

Most of the justification logics still lack tableau proof systems. The aim of this paper is to present tableau proof systems for various justification logics. For each justification logic we present two tableau proof systems. All tableau proof systems are sound and complete with respect to Mkrtychev models of justification logics.

In the first formulation (see Section 3.1), the rules of the tableau system for J is similar to the (J-part) tableau rules given by Renne in [16] for LP. Renne’s tableaux corresponds to the Artemov’s sequent calculus for LP in [2]. The subformula property fails for both the tableaux and the sequent calculus of LP, and also fails for the tableaux of justification logics introduced in this section.

In the second formulation (see Section 3.2), we present a tableau system for JL, which is similar to its KE tableau system but with ordinary propositional rules. Our propositional tableau rules are the ordinary ones given by Smullyan [17], and justification tableau rules are similar to those introduced by Finger [8]. In order to prove the completeness of these tableaux, we give a syntactic proof of cut elimination. Following Finger [8], by restricting the

applications of (*PB*) to analytic ones, we obtain analytic tableaux for justification logics. We give a definition of subformulas in the context of justification logics, and prove that our tableau systems enjoy the subformula property.

2 Justification logics

The language of justification logics is an extension of the language of propositional logic by the formulas of the form $t : F$, where F is a formula and t is a justification term. *Justification terms* (or *terms* for short) are built up from (justification) variables x, y, z, \dots and (justification) constants a, b, c, \dots using several operations depending on the logic: (binary) application ‘ \cdot ’, (binary) sum ‘ $+$ ’, (unary) verifier ‘ $!$ ’, (unary) negative verifier ‘ $?$ ’, and (unary) weak negative verifier ‘ $\bar{?}$ ’. Subterms of a term are defined in the usual way: s is a subterm of $s, s + t, t + s, s \cdot t, !s, \bar{?}s$, and $?s$.

Justification formulas are constructed from a countable set of propositional variables, denoted \mathcal{P} , by the following grammar:

$$A ::= p \mid \perp \mid \neg A \mid A \rightarrow A \mid t : A,$$

where $p \in \mathcal{P}$ and t is a justification term. Other Boolean connectives are defined as usual.

We now begin with describing the axiom schemes and rules of the basic justification logic **J**, and continue with other justification logics. The basic justification logic **J** is the weakest justification logic we shall be discussing. Other justification logics are obtained by adding certain axiom schemes to **J**.

Definition 2.1. *Axioms schemes of **J** are:*

Taut. *All propositional tautologies,*

Sum. $s : A \rightarrow (s + t) : A, \quad s : A \rightarrow (t + s) : A,$

jK. $s : (A \rightarrow B) \rightarrow (t : A \rightarrow (s \cdot t) : B).$

*Other justification logics are obtained by adding the following axiom schemes to **J** in various combinations:*

jT. $t : A \rightarrow A.$

jD. $t : \perp \rightarrow \perp.$

j4. $t : A \rightarrow !t : t : A,$

jB. $\neg A \rightarrow \bar{?}t : \neg t : A.$

j5. $\neg t : A \rightarrow ?t : \neg t : A.$

All justification logics have the inference rule Modus Ponens, and the Iterated Axiom Necessitation rule:

IAN. $\vdash c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : A$, where A is an axiom instance of the logic, c_{i_j} ’s are arbitrary justification constants and $n \geq 1$.

In what follows, **JL** denotes any of the justification logics defined in Definition 2.1, unless stated otherwise. The language of each justification logic **JL** includes those operations on terms that are present in its axioms. $Tm_{\mathbf{JL}}$ and $Fm_{\mathbf{JL}}$ denote the set of all terms and the set of all formulas of **JL** respectively. Moreover, the name of each justification logic is indicated by the list of its axioms. For example, **JT4** is the extension of **J** by axioms **jT** and **j4**, in the language containing term operations $\cdot, +$, and $!$. **JT4** is usually called the logic of proofs **LP**.

Definition 2.2. A constant specification \mathcal{CS} for \mathbf{JL} is a set of formulas of the form $c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : A$, where $n \geq 1$, c_{i_j} 's are justification constants and A is an axiom instance of \mathbf{JL} , such that it is downward closed: if $c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : A \in \mathcal{CS}$, then $c_{i_{n-1}} : \dots : c_{i_1} : A \in \mathcal{CS}$.

The typical form of a formula in a constant specification for \mathbf{JL} is $c : F$, where c is a justification constant, and F is either an axiom instance of \mathbf{JL} or of the form $c_{i_m} : c_{i_{m-1}} : \dots : c_{i_1} : A$, where $m \geq 1$, c_{i_j} 's are justification constants and A is an axiom instance of \mathbf{JL} .

Let $\mathbf{JL}_{\mathcal{CS}}$ be the fragment of \mathbf{JL} where the Iterated Axiom Necessitation rule only produces formulas from the given \mathcal{CS} .

In the remaining of this section, we recall the definitions of M-models for justification logics (see [14, 13]).

Definition 2.3. An M-model $\mathcal{M} = (\mathcal{E}, \mathcal{V})$ for justification logic $\mathbf{JL}_{\mathcal{CS}}$ consists of a valuation $\mathcal{V} : \mathcal{P} \rightarrow \{0, 1\}$ and an admissible evidence function $\mathcal{E} : Tm_{\mathbf{JL}} \rightarrow 2^{Fm_{\mathbf{JL}}}$ meeting the following conditions:

- E1. $A \rightarrow B \in \mathcal{E}(s)$ and $A \in \mathcal{E}(t)$ implies $B \in \mathcal{E}(s \cdot t)$.
- E2. $\mathcal{E}(s) \cup \mathcal{E}(t) \subseteq \mathcal{E}(s + t)$.
- E3. $c : F \in \mathcal{CS}$ implies $F \in \mathcal{E}(c)$.

Definition 2.4. For an M-model $\mathcal{M} = (\mathcal{E}, \mathcal{V})$ the forcing relation \Vdash is defined as follows:

1. $\mathcal{M} \not\Vdash \perp$,
2. $\mathcal{M} \Vdash p$ iff $\mathcal{V}(p) = 1$, for $p \in \mathcal{P}$,
3. $\mathcal{M} \Vdash \neg A$ iff $\mathcal{M} \not\Vdash A$,
4. $\mathcal{M} \Vdash A \rightarrow B$ iff $\mathcal{M} \not\Vdash A$ or $\mathcal{M} \Vdash B$,
5. If \mathbf{JL} does not contain axiom jT : $\mathcal{M} \Vdash t : A$ iff $A \in \mathcal{E}(t)$.
If \mathbf{JL} contains axiom jT : $\mathcal{M} \Vdash t : A$ iff $A \in \mathcal{E}(t)$ and $\mathcal{M} \Vdash A$.

If $\mathcal{M} \Vdash F$ then it is said that F is true in \mathcal{M} or \mathcal{M} satisfies F .

In order to define M-models for other justification logics of Definition 2.1 certain additional conditions should be imposed on the M-model.

Definition 2.5. An M-model $\mathcal{M} = (\mathcal{E}, \mathcal{V})$ for justification logic $\mathbf{JL}_{\mathcal{CS}}$ is an M-model for $\mathbf{JL}_{\mathcal{CS}}$ such that:

- if \mathbf{JL} contains axiom jD , then for all $t \in Tm_{\mathbf{JL}}$:
E4. $\perp \notin \mathcal{E}(t)$.
- if \mathbf{JL} contains axiom $j4$, then for all $t \in Tm_{\mathbf{JL}}$ and $A \in Fm_{\mathbf{JL}}$:
E5. $A \in \mathcal{E}(t)$ implies $t : A \in \mathcal{E}(!t)$.
- if \mathbf{JL} contains axiom jB , then for all $t \in Tm_{\mathbf{JL}}$ and $A \in Fm_{\mathbf{JL}}$:
E6. $\mathcal{M} \not\Vdash A$ implies $\neg t : A \in \mathcal{E}(?t)$.
- if \mathbf{JL} contains axiom $j5$, then for all $t \in Tm_{\mathbf{JL}}$ and $A \in Fm_{\mathbf{JL}}$:
E7. $A \notin \mathcal{E}(t)$ implies $\neg t : A \in \mathcal{E}(?t)$.

By a $\mathbf{JL}_{\mathcal{CS}}$ -model we mean an M-model for justification logic $\mathbf{JL}_{\mathcal{CS}}$. A \mathbf{JL} -formula F is $\mathbf{JL}_{\mathcal{CS}}$ -valid if it is true in every $\mathbf{JL}_{\mathcal{CS}}$ -model. For a set S of formulas, $\mathcal{M} \Vdash S$ provided that $\mathcal{M} \Vdash F$ for all formulas F in S . Note that given a constant specification \mathcal{CS} for \mathbf{JL} , and a model \mathcal{M} of $\mathbf{JL}_{\mathcal{CS}}$ we have $\mathcal{M} \Vdash \mathcal{CS}$ (in this case it is said that \mathcal{M} respects \mathcal{CS}).

The proof of soundness and completeness theorems for all justification logics of Definition 2.1 are given in [12, 13].

Theorem 2.1. Let \mathbf{JL} be one of the justification logics of Definition 2.1, and \mathcal{CS} be a constant specification for \mathbf{JL} . Then a \mathbf{JL} -formula F is provable in $\mathbf{JL}_{\mathcal{CS}}$ iff F is $\mathbf{JL}_{\mathcal{CS}}$ -valid.

3 Tableaux

In this section we present two different tableau proof systems for each justification logic of Definition 2.1. The rules of our first tableau system for J in Section 3.1 is similar to that given in [16]. In Section 3.2 we present a tableau system for J which is similar to the KE tableau system of LP in [8], but with ordinary propositional rules instead of linear propositional KE rules.

3.1 JL-Tableaux

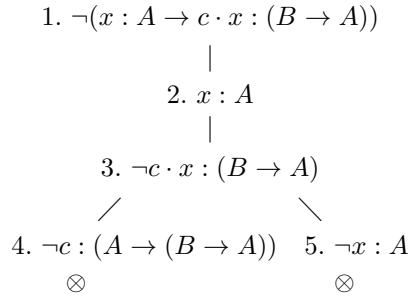
Tableau proof systems for the logic of proofs are given in [10, 15, 16]. In this section we present similar tableaux for all justification logics.

A J_{CS} -tableau for a formula is a binary tree with the negation of that formula at the root constructed by applying J_{CS} -tableau rules from Table 1. For extensions of J , tableau rules corresponding to axioms from Table 2 should be added to J_{CS} -tableau rules. For example, the tableau proof system of the logic of proofs LP is obtained by adding the rules $(T :)$ and $(F!)$ to the tableau rules of J . For a justification logic JL, a tableau branch of a JL_{CS} -tableau closes if one of the following holds:

1. Both A and $\neg A$ occurs in the branch, for some formula A .
2. \perp occurs in the branch.
3. $\neg c : F$ occurs in the branch, for some $c : F \in CS$.

A tableau closes if all branches of the tableau close. A JL_{CS} -tableau proof for formula F is a closed tableau beginning with $\neg F$ (the root of the tableau) using only tableau rules of JL_{CS} . A JL_{CS} -tableau for a finite set S of JL-formulas begins with a single branch whose nodes consist of the formulas of S as roots.

Example 3.1. We give a J_{CS} -tableau proof of $x : A \rightarrow c \cdot x : (B \rightarrow A)$, where CS contains $c : (A \rightarrow (B \rightarrow A))$.



Formulas 2 and 3 are from 1 by rule $(F \rightarrow)$, and 4 and 5 are from 3 by rule $(F \cdot)$. Closed branches are indicated by \otimes .

Let us show the soundness and completeness of tableau systems with respect to M-models. Our starting point is the following lemma, whose proof is straightforward and is omitted here.

Lemma 3.1. *Let π be any branch of a JL_{CS} -tableau and \mathcal{M} be a JL_{CS} -model that satisfies all the formulas occur in π . If a JL_{CS} -tableau rule is applied to π , then it produces at least one extension π' such that \mathcal{M} satisfies all the formulas occur in π' .*

Propositional rules:	
$\frac{\neg\neg A}{A} (F\neg)$	
$\frac{\neg(A \rightarrow B)}{A} (F \rightarrow)$	$\frac{A \rightarrow B}{\neg A B} (T \rightarrow)$
$\neg B$	
Justification rules:	
$\frac{\neg t + s : A}{\neg t : A} (F+)$	$\frac{\neg s \cdot t : B}{\neg s : (A \rightarrow B) \neg t : A} (F\cdot)$
$\neg s : A$	

Table 1. Tableau rules for basic justification logic J.

Justification axiom	Tableau rule
jT. $t : A \rightarrow A$	$\frac{t : A}{A} (T :)$
jD. $t : \perp \rightarrow \perp$	$\frac{}{\neg t : \perp} (F : \perp)$
j4. $t : A \rightarrow !t : t : A$	$\frac{\neg !t : t : A}{\neg t : A} (F!)$
jB. $\neg A \rightarrow \bar{?}t : \neg t : A$	$\frac{\neg \bar{?}t : \neg t : A}{A} (F\bar{?})$
j5. $\neg t : A \rightarrow ?t : \neg t : A$	$\frac{\neg ?t : \neg t : A}{t : A} (F?)$

Table 2. Justification axioms with corresponding tableau rules.

Theorem 3.1 (Soundness). *If A has a JL_{CS} -tableau proof, then it is JL_{CS} -valid.*

Proof. If A is not JL_{CS} -valid, then there is a JL_{CS} -model \mathcal{M} such that $\mathcal{M} \Vdash \neg A$. Thus by Lemma 3.1, there is no closed JL_{CS} -tableau beginning with $\neg A$. Therefore, A does not have a JL_{CS} -tableau proof. \square

Next we shall prove the completeness theorem, by making use of maximal consistent sets.

Definition 3.1. *Suppose Γ is a set of JL-formulas. Γ is (tableau) JL_{CS} -consistent if there is no closed tableau beginning with any finite subset of Γ . Γ is maximal if it has no proper tableau consistent extension.*

It is known that every JL_{CS} -consistent set has a maximally JL_{CS} -consistent extension (Lindenbaum Lemma).

It is easy to show that maximally JL_{CS} -consistent sets are closed under JL_{CS} -tableau rules. For a non-branching rule like

$$\frac{\alpha}{\alpha_1 \quad \alpha_2}$$

this means that if α is in a maximally JL_{CS} -consistent set Γ , then both $\alpha_1 \in \Gamma$ and $\alpha_2 \in \Gamma$. For a branching rule like

$$\frac{\beta}{\beta_1 | \beta_2}$$

this means that if β is in a maximally JL_{CS} -consistent set Γ , then $\beta_1 \in \Gamma$ or $\beta_2 \in \Gamma$. For the rule $(F\cdot)$ this means that if $\neg s \cdot t : B \in \Gamma$, then for every formula A either $\neg s : (A \rightarrow B) \in \Gamma$ or $\neg t : A \in \Gamma$.

Lemma 3.2. *Suppose Γ is a maximally JL_{CS} -consistent set. Then Γ is closed under JL_{CS} -tableau rules.*

Proof. The proof for propositional rules $(F\neg)$, $(F\rightarrow)$, and $(T\rightarrow)$ are standard. For justification rules, we detail the proof only for the rules $(F\cdot)$ and $(F:\perp)$. The proof for the other tableau justification rules is similar.

For $(F\cdot)$, suppose Γ is a maximally JL_{CS} -consistent set and $\neg s \cdot t : B \in \Gamma$. Suppose towards a contradiction that for some formula A we have $\neg s : (A \rightarrow B) \notin \Gamma$ and $\neg t : A \notin \Gamma$. Since Γ is maximal, we have $\Gamma \cup \{\neg s : (A \rightarrow B)\}$ and $\Gamma \cup \{\neg t : A\}$ are not tableau JL_{CS} -consistent. Thus there are closed JL_{CS} -tableaux for finite subsets, say $\Gamma_1 \cup \{\neg s : (A \rightarrow B)\}$ and $\Gamma_2 \cup \{\neg t : A\}$. But $\Gamma_1 \cup \Gamma_2 \cup \{\neg s \cdot t : B\}$ is a finite subset of Γ and, using rule $(F\cdot)$, there is a closed JL_{CS} -tableau for it, contra the tableau JL_{CS} -consistency of Γ .

For $(F:\perp)$, suppose towards a contradiction that $\neg t : \perp \notin \Gamma$, for some term t . Then, $\Gamma \cup \{\neg t : \perp\}$ is not tableau JL_{CS} -consistent. Thus, there is a closed JL_{CS} -tableau for a finite subset, say $\Gamma_0 \cup \{\neg t : \perp\}$. Using rule $(F:\perp)$, there is a closed JL_{CS} -tableau for Γ_0 , contra the tableau JL_{CS} -consistency of Γ . Therefore, $\neg t : \perp \in \Gamma$, for any term t . \square

Definition 3.2. *Given a maximally JL_{CS} -consistent set Γ , the canonical model $\mathcal{M} = (\mathcal{E}, \mathcal{V})$ with respect to Γ is defined as follows:*

- $\mathcal{E}(t) = \{A \mid \neg t : A \notin \Gamma\}$.
- $\mathcal{V}(p) = 1$ iff $p \in \Gamma$, where $p \in \mathcal{P}$.

Lemma 3.3 (Truth Lemma). *Suppose Γ is a maximally JL_{CS} -consistent set and $\mathcal{M} = (\mathcal{E}, \mathcal{V})$ is the canonical model with respect to Γ . Then for every JL -formula F :*

1. $F \in \Gamma$ implies $\mathcal{M} \Vdash F$.
2. $\neg F \in \Gamma$ implies $\mathcal{M} \nVdash F$.

Proof. By induction on the complexity of F . The base case and the propositional inductive cases are standard. The proof for the case that $F = t : A$ is as follows.

Suppose that $t : A \in \Gamma$. Since Γ is tableau JL_{CS} -consistent, $\neg t : A \notin \Gamma$. Thus $A \in \mathcal{E}(t)$. If JL does not contain axiom jT , then $\mathcal{M} \Vdash t : A$ as desired. If JL contains axiom jT , then since Γ is closed under $(T:)$, $A \in \Gamma$. Thus, by the induction hypothesis, $\mathcal{M} \Vdash A$. Hence $\mathcal{M} \Vdash t : A$.

Suppose that $\neg t : A \in \Gamma$. Thus $A \notin \mathcal{E}(t)$, and hence $\mathcal{M} \nVdash t : A$. \square

Lemma 3.4. *Given a maximally JL_{CS} -consistent set Γ , the canonical model $\mathcal{M} = (\mathcal{E}, \mathcal{V})$ with respect to Γ is a JL_{CS} -model.*

Proof. Suppose Γ is a maximally $\text{JL}_{\mathcal{CS}}$ -consistent set and $\mathcal{M} = (\mathcal{E}, \mathcal{V})$ is the canonical model with respect to Γ . We shall show that the admissible evidence function \mathcal{E} satisfies the corresponding conditions stated in the definition of $\text{JL}_{\mathcal{CS}}$ -models.

For $\mathcal{E}1$, suppose that $A \in \mathcal{E}(t)$ and $A \rightarrow B \in \mathcal{E}(s)$. We have to show that $B \in \mathcal{E}(s \cdot t)$. By the definition of \mathcal{E} , $\neg t : A \notin \Gamma$ and $\neg s : (A \rightarrow B) \notin \Gamma$. By Lemma 3.2, Γ is closed under rule $(F\cdot)$, and hence $\neg s \cdot t : B \notin \Gamma$. Hence, by the definition of \mathcal{E} , $B \in \mathcal{E}(s \cdot t)$.

For $\mathcal{E}2$, suppose that $A \in \mathcal{E}(s) \cup \mathcal{E}(t)$. We have to show that $A \in \mathcal{E}(s + t)$. If $A \in \mathcal{E}(s)$, then $\neg s : A \notin \Gamma$. By Lemma 3.2, Γ is closed under rule $(F+)$, and hence $\neg s + t : A \notin \Gamma$. Therefore, $A \in \mathcal{E}(s + t)$. The case that $A \in \mathcal{E}(t)$ is similar.

For $\mathcal{E}3$, suppose that $c : F \in \mathcal{CS}$. We have to show that $F \in \mathcal{E}(c)$. Since Γ is $\text{JL}_{\mathcal{CS}}$ -consistent, $\neg c : F \notin \Gamma$. Thus $F \in \mathcal{E}(c)$.

For $\mathcal{E}4$, where JL contains axiom jD , by Lemma 3.2 we have $\neg t : \perp \in \Gamma$ for any term $t \in \text{Term}_{\text{JL}}$. Thus $\perp \notin \mathcal{E}(t)$.

For $\mathcal{E}5$, where JL contains axiom j4 , suppose that $A \in \mathcal{E}(t)$. We have to show that $t : A \in \mathcal{E}(!t)$. By the definition of \mathcal{E} , $\neg t : A \notin \Gamma$. By Lemma 3.2, Γ is closed under rule $(F!)$, and hence $\neg !t : t : A \notin \Gamma$. Therefore, $t : A \in \mathcal{E}(!t)$.

For $\mathcal{E}6$, where JL contains axiom jB , suppose that $\mathcal{M} \not\models A$. We have to show that $\neg t : A \in \mathcal{E}(\bar{?}t)$. By the Truth Lemma, $A \notin \Gamma$. By Lemma 3.2, Γ is closed under rule $(F\bar{?})$, and hence $\neg \bar{?}t : \neg t : A \notin \Gamma$. Therefore, $\neg t : A \in \mathcal{E}(\bar{?}t)$.

For $\mathcal{E}7$, where JL contains axiom j5 , suppose that $A \notin \mathcal{E}(t)$. We have to show that $\neg t : A \in \mathcal{E}(?t)$. By the definition of \mathcal{E} , $\neg t : A \in \Gamma$. By Lemma 3.2, Γ is closed under rule $(F?)$, and hence $\neg ?t : \neg t : A \notin \Gamma$. Therefore, $\neg t : A \in \mathcal{E}(?t)$. \square

Theorem 3.2 (Completeness). *If A is $\text{JL}_{\mathcal{CS}}$ -valid, then it has a $\text{JL}_{\mathcal{CS}}$ -tableau proof.*

Proof. If A does not have a $\text{JL}_{\mathcal{CS}}$ -tableau proof, then $\{\neg A\}$ is a $\text{JL}_{\mathcal{CS}}$ -consistent set and can be extended to a maximal $\text{JL}_{\mathcal{CS}}$ -consistent set Γ . Since $\neg A \in \Gamma$, by the Truth Lemma, $\mathcal{M} \not\models A$, where \mathcal{M} is the canonical model of $\text{JL}_{\mathcal{CS}}$ with respect to Γ . Therefore A is not $\text{JL}_{\mathcal{CS}}$ -valid. \square

Clearly in any $\text{JL}_{\mathcal{CS}}$ -tableau system the rule $(F\cdot)$

$$\frac{\neg s \cdot t : B}{\neg s : (A \rightarrow B) | \neg t : A} (F\cdot)$$

is not analytic, because the formula A in the conclusion of the rule could be a new formula from the outside of the proof. The rule $(F : \perp)$ is not analytic too. In the following section we replace these rules with analytic rules.

3.2 $\text{JL}^{\mathcal{T}}$ -tableaux

In this section we present analytic tableaux for justification logics. The rule $(F\cdot)$ is replaced with the analytic non-branching rule $(T\cdot)$ (see Table 3) and $(F : \perp)$ is replaced with an analytic rule. The rule $(T\cdot)$ was introduced by Finger in [8] in a tableau proof system for the logic of proofs based on KE tableaux (cf. [5, 6, 7]).¹ The tableau proof system of this section

¹ It is worth noting that Finger's completeness proof of KE tableaux for the logic of proofs in [8] contains a mistake. In fact, he wrongly claimed that every LP-tableau proof (see Section 3.1) can be simulated by KE tableaux of LP. Then he used the completeness of LP-tableaux to show that KE tableau system of LP is complete.

is similar to KE tableaux, with the difference that its propositional logic rules is the same as Smullyan's rules [17]. A restricted form of the cut rule, called the principle of bivalence in [5, 6, 7] and denoted by (PB) , is also added to the rules. In order to make the rules $(T\cdot)$ and (PB) analytic we put some restrictions on the application of these rules. Let us first extend the definition of subformulas of a formula to include constant specifications.

Definition 3.3. Let \mathcal{CS} be a constant specification for \mathbf{JL} , and let A and B be \mathbf{JL} -formulas. A is a $\mathbf{JL}_{\mathcal{CS}}$ -subformula of B if one of the following clauses holds:

1. $A = B$,
2. $B = \neg F$, and A is a $\mathbf{JL}_{\mathcal{CS}}$ -subformula of F ,
3. $B = F \rightarrow G$, and A is a $\mathbf{JL}_{\mathcal{CS}}$ -subformula of F or G ,
4. $B = t : F$, and A is a $\mathbf{JL}_{\mathcal{CS}}$ -subformula of F ,
5. $A = t : F$, where t is a subterm of a term in B and F is a $\mathbf{JL}_{\mathcal{CS}}$ -subformula of B ,
6. A is a $\mathbf{JL}_{\mathcal{CS}}$ -subformula of $c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : F \in \mathcal{CS}$, where F is an axiom instance of \mathbf{JL} .
7. The relation of “ $\mathbf{JL}_{\mathcal{CS}}$ -subformula of”, defined in clauses 1-6, is extended by transitivity.

A is a weak $\mathbf{JL}_{\mathcal{CS}}$ -subformula of B if A is either a $\mathbf{JL}_{\mathcal{CS}}$ -subformula of B or the negation of a $\mathbf{JL}_{\mathcal{CS}}$ -subformula of B .

Tableau rules for basic justification logic \mathbf{J} are given in Table 3. We denote this tableau system by $\mathbf{J}^{\mathcal{T}}$. For extensions of \mathbf{J} , tableau rules corresponding to axioms from Table 2 should be added to the rules of $\mathbf{J}^{\mathcal{T}}$, except that in those justification logics that contain axiom \mathbf{jD} the rule $(F : \perp)$ is replaced by the following rule:

$$\frac{t : \perp}{\perp} (T : \perp)$$

The closure conditions are the same as \mathbf{JL} -tableaux. For a justification logic \mathbf{JL} , the resulting tableau system is denoted by $\mathbf{JL}^{\mathcal{T}}$.

Note that in $\mathbf{JL}^{\mathcal{T}}$ -tableaux the rules $(T\cdot)$ and (PB) have restrictions on their applications (see Table 3). The formula A in the conclusion of (PB) is called the PB -formula. Furthermore, the rule $(T\cdot)$ is a binary rule (it takes two formulas as input), and it should be read as follows: if a branch contains $s : (A \rightarrow B)$ and $t : A$, then we can extend that branch by adding $s \cdot t : B$, provided that the formulas $s : (A \rightarrow B)$, $t : A$, and $s \cdot t : B$ are all $\mathbf{JL}_{\mathcal{CS}}$ -subformulas of the root of the tableau. In addition, there is no ordering intended on the input $s : (A \rightarrow B)$, $t : A$.

From Definition 3.3 it is obvious that the following is an instance of (PB) :

$$\frac{c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : A \mid \neg c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : A}{\phantom{c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : A}} (PB)$$

where $c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : A \in \mathcal{CS}$. Since the right branch is closed, it follows that the following rule is admissible in $\mathbf{JL}_{\mathcal{CS}}^{\mathcal{T}}$:

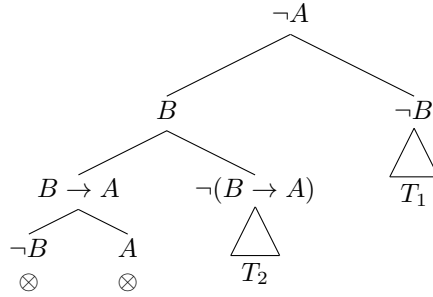
$$\frac{}{c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : A}$$

where $c_{i_n} : c_{i_{n-1}} : \dots : c_{i_1} : A \in \mathcal{CS}$.

The cut rule is the same as the principle of bivalence (PB) but without any restrictions on the cut-formula A . Completeness is proved by first showing that all theorems of $\mathbf{JL}_{\mathcal{CS}}$ are provable in the tableau system $\mathbf{JL}_{\mathcal{CS}}^{\mathcal{T}} + (cut)$, and then by proving the cut elimination theorem for $\mathbf{JL}_{\mathcal{CS}}^{\mathcal{T}} + (cut)$.

Theorem 3.4 (Completeness). *If A is provable in $\mathbf{JL}_{\mathcal{CS}}$, then it is provable in the tableau system $\mathbf{JL}_{\mathcal{CS}}^{\mathcal{T}} + (cut)$.*

Proof. The proof is by induction on the proof of A in $\mathbf{JL}_{\mathcal{CS}}$. It is a routine matter to check that all axioms of \mathbf{JL} are provable in $\mathbf{JL}_{\mathcal{CS}}^{\mathcal{T}}$, even without using (PB) and (cut). If A is obtained from B and $B \rightarrow A$ by MP, then by the induction hypothesis there are closed $\mathbf{JL}_{\mathcal{CS}}^{\mathcal{T}}$ -tableaux T_1 and T_2 for B and $B \rightarrow A$ respectively. Then, using the cut rule twice, the following is a closed tableau for A



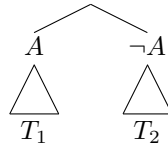
Finally, if $A = c : F \in \mathcal{CS}$ is obtained by IAN, then by the closure condition $\neg c : F$ is a closed one-node tableau. \square

The proof of the cut elimination is similar to the algorithm given by Fitting in [9], and thus the details will be omitted. The following definitions are inspired from those in [9].

Definition 3.4. *The rank of a term t and a formula A , denoted by $r(t)$ and $r(A)$ respectively, is defined inductively as follows:*

1. $r(x) = r(c) = 0$, for justification variable x and justification constant c ,
 $r(s + t) = r(s \cdot t) = r(s) + r(t) + 1$, $r(!t) = r(?t) = r(t) + 1$.
2. $r(p) = r(\perp) = 0$, for $p \in \mathcal{P}$,
 $r(\neg A) = r(A) + 1$, $r(A \rightarrow B) = r(A) + r(B) + 1$, $r(t : A) = r(t) + r(A) + 1$.

Definition 3.5. *Suppose that in a tableau T there is a cut to A and $\neg A$ of the following form:*



where T_1 and T_2 are the subtableaux below A and $\neg A$, respectively. Let $|T|$ denote the number of formulas in the tableau T .

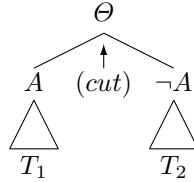
1. We say the cut is at a branch end if $|T_1| = 0$ or $|T_2| = 0$; that is, if either there are no formulas below A , or there are no formulas below $\neg A$, or both.

2. The rank of the cut is the rank of the cut-formula A .
3. The weight of the cut is the number of formulas in T strictly below A and $\neg A$; that is, the weight of the cut is $|T_1| + |T_2|$.
4. The cut is called minimal if there are no cuts in the subtableaux T_1 and T_2 .

The following fact will be used frequently in the proof of cut elimination (cf. [9]). Suppose that T is a closed tableau for a finite set S of formulas and $S \subseteq S'$, where S' is also finite. Then there is a closed tableau for S' with the same number of steps.

Theorem 3.5 (Cut Elimination). *If a formula is provable in the tableau system $\text{JL}_{\mathcal{CS}}^T + (\text{cut})$, then it is also provable in $\text{JL}_{\mathcal{CS}}^T$.*

Proof. We will show how to eliminate the minimal cuts from a tableau T . Suppose T consists a minimal cut of the following form:



The proof is by induction on the rank of the cut-formula A with subinduction on the weight of the cut. Similar to the cut elimination of the sequent calculus of classical logic (cf. [18]), we distinguish three cases:

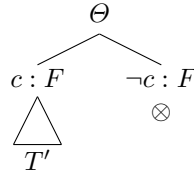
Case I. The minimal cut is at a branch end.

Case II. The minimal cut is not at a branch end, and the uppermost formulas in T_1 or T_2 are obtained by applying a tableau rule to a formula from Θ .

Case III. The minimal cut is not at a branch end, and the uppermost formulas in T_1 and T_2 are obtained by applying tableau rules to A and $\neg A$, respectively.

In case I, we eliminate the minimal cut. In cases II and III, we transform the tableau T into another closed tableau in which the minimal cut is replaced by cuts of lower rank, by cuts of the same rank but of lower weight, or both.

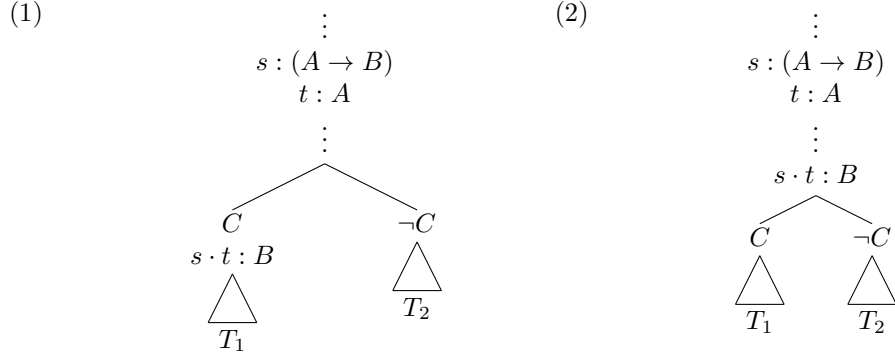
Case I. Suppose we have a minimal cut at the end of a branch. We only consider the case in which the branch closes because of $\neg c : F$, where $c : F \in \mathcal{CS}$ (see [9] for the other cases). In this case the cut looks like this.



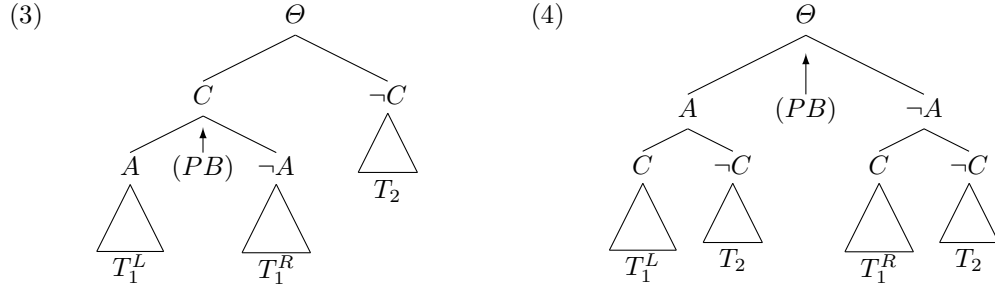
Since $c : F \in \mathcal{CS}$, the cut-formula $c : F$ is a $\text{JL}_{\mathcal{CS}}$ -subformula of the root, and hence the cut is an instance of (PB) .

Case II. Suppose the minimal cut is not at a branch end, and the uppermost formulas in T_1 or T_2 are obtained by applying a tableau rule to formulas from Θ . In this case we push the cut down in the tableau and obtain a new cut of lower weight. We only consider two cases: (i) the rule $(T\cdot)$ is applied to formulas from Θ , and (ii) the rule (PB) is applied. The other cases are similar.

Suppose the rule $(T\cdot)$ is applied to formulas from Θ . Then the cut is of the form shown in (1), where $s : (A \rightarrow B)$, $t : A$, and $s \cdot t : B$ are $\mathbf{JL}_{\mathcal{CS}}$ -subformulas of the root. The displayed cut in (1) is transformed into the one in (2) of lower weight.

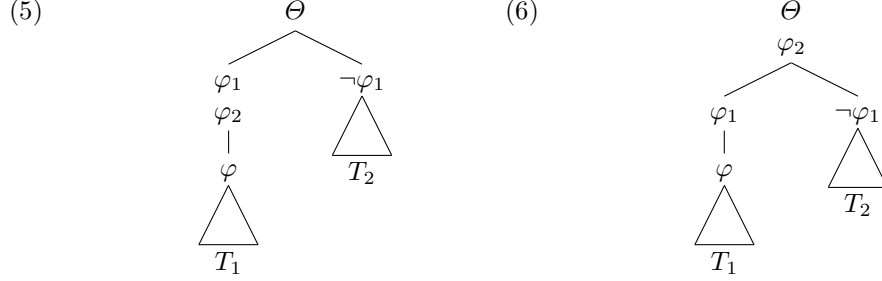


Now suppose the rule (PB) is applied. Then the cut is of the form shown in (3), where A is a $\mathbf{JL}_{\mathcal{CS}}$ -subformula of the root. The displayed cut in (3) is transformed into the one in (4) of lower weight.

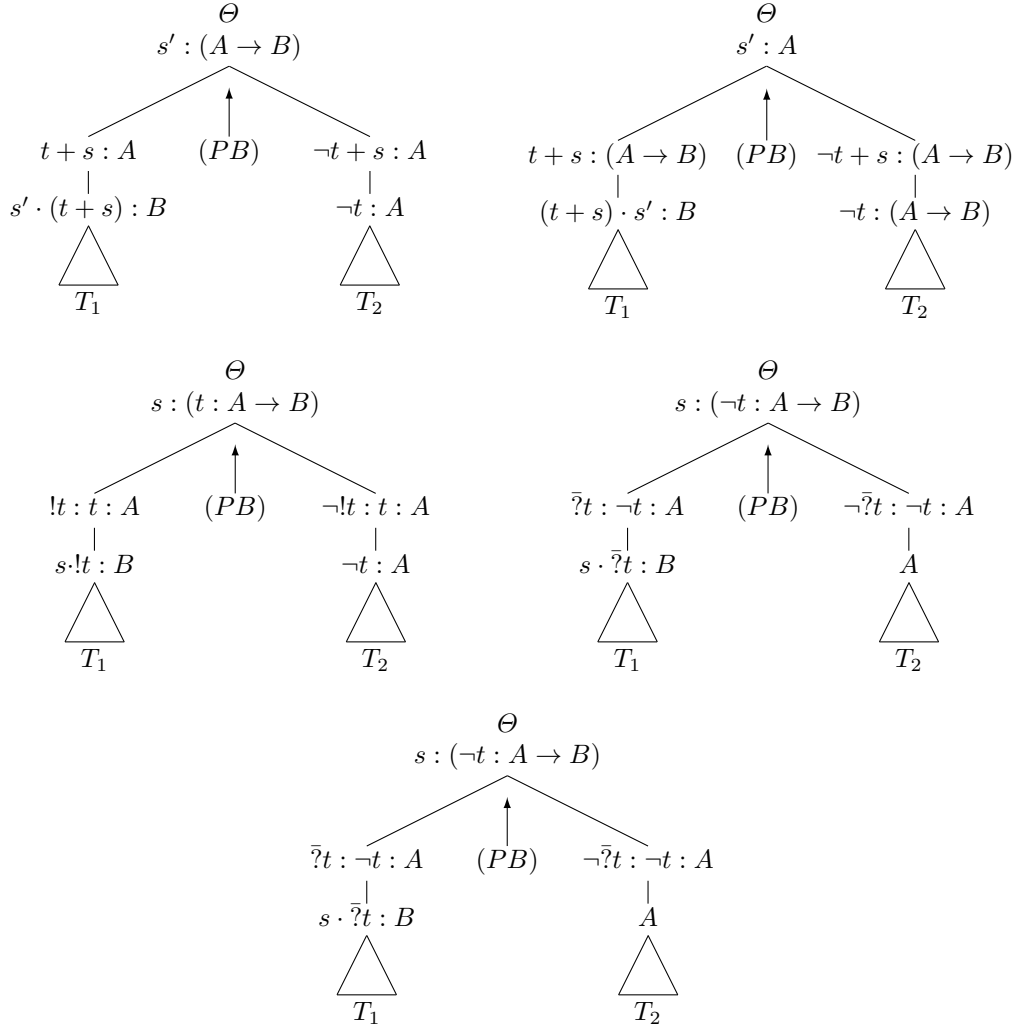


Case III. Suppose the minimal cut is not at a branch end, and the uppermost formulas in T_1 and T_2 are obtained by applying tableau rules to A and $\neg A$, respectively. In this case we transform the cut into cuts of lower rank, or into cuts with the same rank but of lower weight.

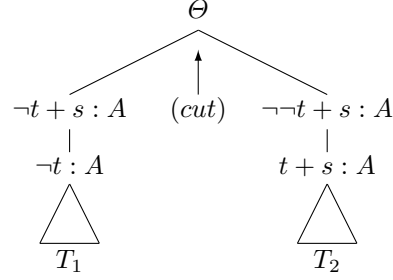
First consider the rule $(T\cdot)$ which is a two-premised rule of the form $\frac{\varphi_1}{\frac{\varphi_2}{\varphi}}$. Since φ_1 is a JL_{CS} -subformula of the root, the two cuts to φ_1 and $\neg\varphi_1$ shown in (5) and (6) are instances of (PB) . The same holds if in (5) or (6) a cut to φ_2 and $\neg\varphi_2$ is applied.



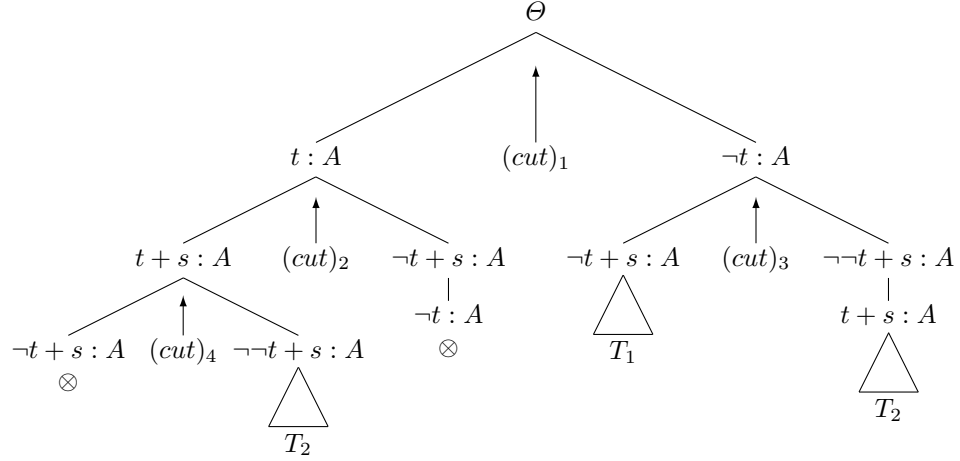
For example, the following cuts are instances of (PB) .



Consider the following cut to formulas $\neg t + s : A$ and $\neg\neg t + s : A$ to which the rules $(F+_L)$ and $(F\neg)$ are applied respectively.

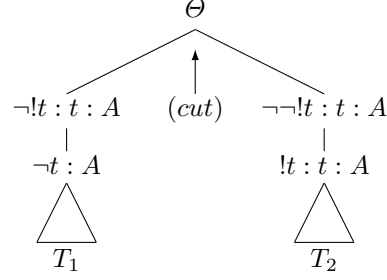


This cut is transformed into the following cuts.

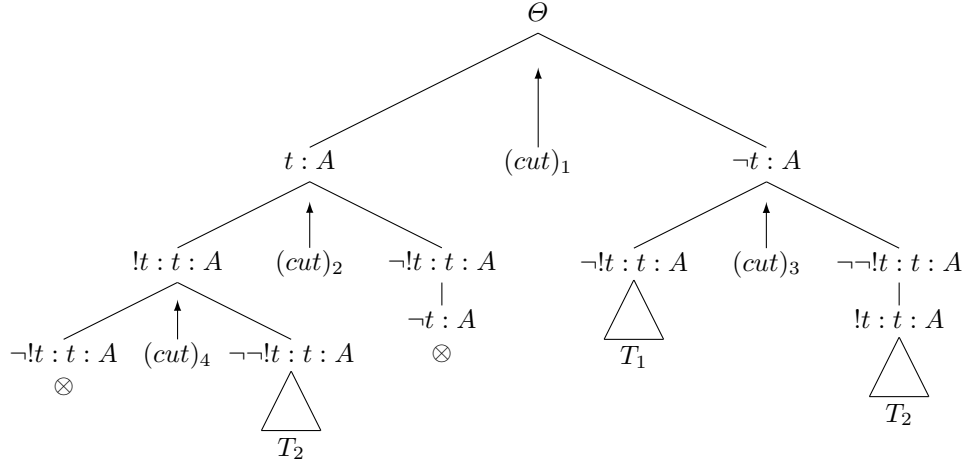


The rank of $(cut)_1$ and $(cut)_2$ is less than the rank of (cut) . Moreover, $(cut)_3$ and $(cut)_4$ have the same rank as (cut) but their weight are smaller than the weight of (cut) . The case of $(F+_R)$ is treated in a similar way.

Consider the following cut to formulas $\neg!t : t : A$ and $\neg\neg!t : t : A$ to which the rules $(F!)$ and $(F\neg)$ are applied respectively.

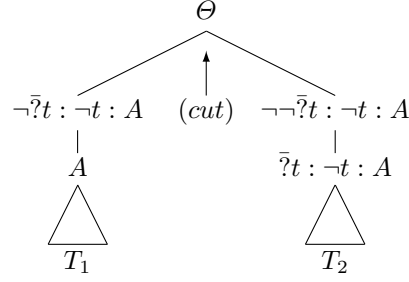


This cut is transformed into the following cuts.

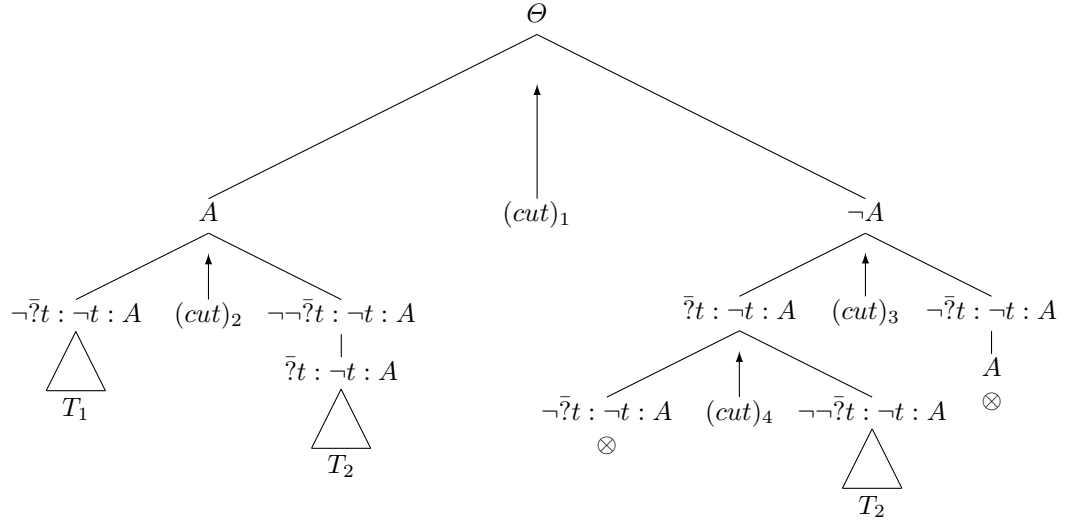


The rank of $(cut)_1$ and $(cut)_2$ is less than the rank of (cut) . Moreover, $(cut)_3$ and $(cut)_4$ have the same rank as (cut) but their weight are smaller than the weight of (cut) . The cut to formulas $\neg?t : \neg t : A$ and $\neg\neg?t : \neg t : A$ to which the rules $(F?)$ and $(F\neg)$ are applied respectively is treated similarly.

Consider the following cut to formulas $\neg\bar{?}t : \neg t : A$ and $\neg\neg\bar{?}t : \neg t : A$ to which the rules $(F\bar{?})$ and $(F\neg)$ are applied respectively.

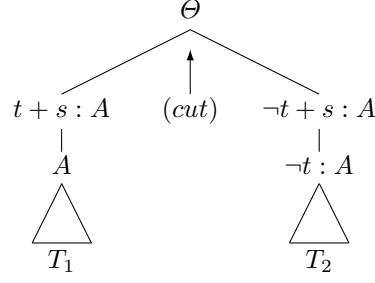


This cut is transformed into the following cuts.

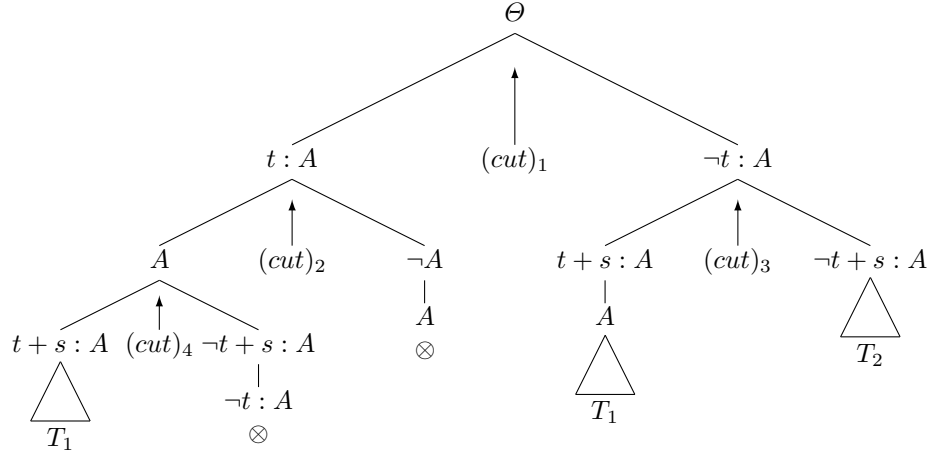


The rank of $(cut)_1$ and $(cut)_3$ is less than the rank of (cut) . Moreover, $(cut)_2$ and $(cut)_4$ have the same rank as (cut) but their weight are smaller than the weight of (cut) .

Now suppose that jT is an axiom of JL . Consider the following cut to formulas $t + s : A$ and $\neg t + s : A$ to which the rules $(T :)$ and $(F+_L)$ are applied respectively.

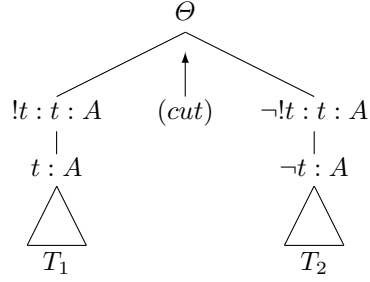


This cut is transformed into the following cuts.

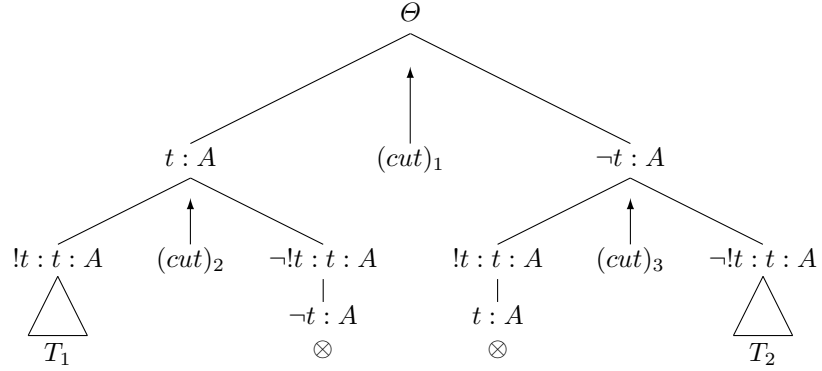


The rank of $(cut)_1$ and $(cut)_2$ is less than the rank of (cut) . Moreover, $(cut)_3$ and $(cut)_4$ have the same rank as (cut) but their weight are smaller than the weight of (cut) . The case of $(F+_R)$ is similar.

Consider the following cut to formulas $!t : t : A$ and $\neg !t : t : A$ to which the rules $(T :)$ and $(F!)$ are applied respectively.

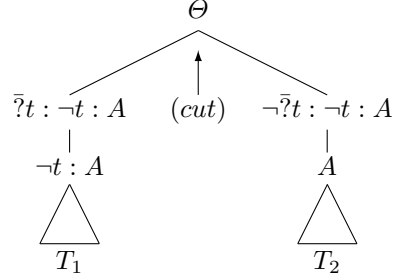


This cut is transformed into the following cuts.

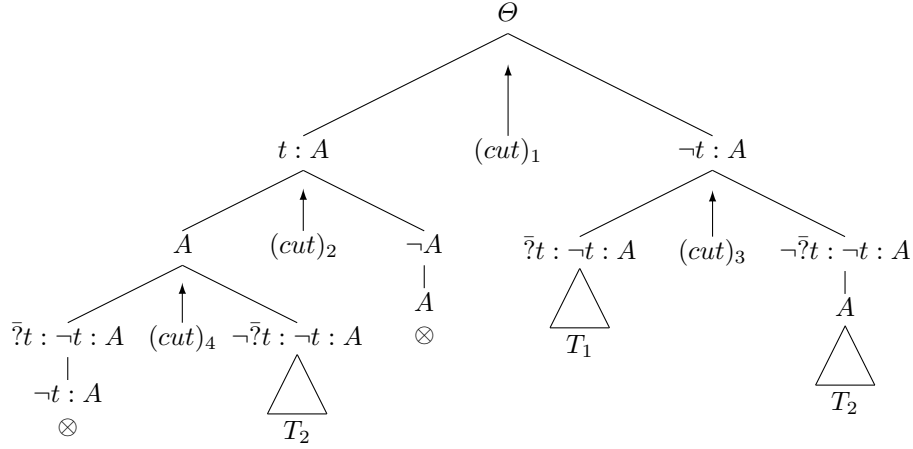


The rank of $(cut)_1$ is less than the rank of (cut) . Moreover, $(cut)_2$ and $(cut)_3$ have the same rank as (cut) but their weight are smaller than the weight of (cut) . The cut to formulas $?t : \neg t : A$ and $\neg ?t : \neg t : A$ to which the rules $(T :)$ and $(F?)$ are applied respectively is treated similarly.

Consider the following cut to formulas $\bar{?}t : \neg t : A$ and $\neg \bar{?}t : \neg t : A$ to which the rules $(T :)$ and $(F \bar{?})$ are applied respectively.

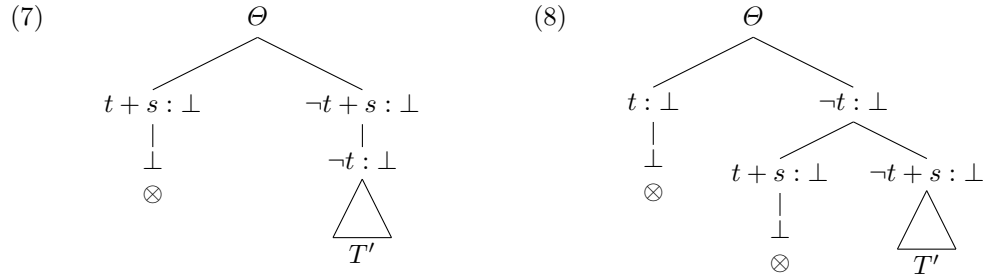


This cut is transformed into the following cuts.



The rank of $(cut)_1$ and $(cut)_2$ is less than the rank of (cut) . Moreover, $(cut)_3$ and $(cut)_4$ have the same rank as (cut) but their weight are smaller than the weight of (cut) .

Consider the cut to formulas $t + s : \perp$ and $\neg t + s : \perp$, shown in (7), to which the rules $(T : \perp)$ and $(F +_L)$ are applied respectively.



The cut in (7) is transformed into the cuts shown in (8), in which the cut to $t : \perp$ and $\neg t : \perp$ has a lower rank, and the weight of the cut to $t + s : \perp$ and $\neg t + s : \perp$ is smaller than the weight of the original cut. The case of $(F +_R)$ is treated in a similar way.

Actually there are two remaining cuts to verify in this case: the cut to formulas $A \rightarrow B$ and $\neg(A \rightarrow B)$ to which the rules $(T \rightarrow)$ and $(F \rightarrow)$ are applied respectively; and the cut

to formulas $\neg(A \rightarrow B)$ and $\neg\neg(A \rightarrow B)$ to which the rules $(F \rightarrow)$ and $(F\neg)$ are applied respectively. We refer the reader to [9] for a more detailed exposition of these two cuts. \square

Theorem 3.6 (Completeness). *If A is $\text{JL}_{\mathcal{CS}}$ -valid, then it has a $\text{JL}_{\mathcal{CS}}^{\mathcal{T}}$ -tableau proof.*

Proof. If A is $\text{JL}_{\mathcal{CS}}$ -valid, then by Theorem 2.1 it is provable in $\text{JL}_{\mathcal{CS}}$. Hence, by Theorem 3.4, it is provable in $\text{JL}_{\mathcal{CS}}^{\mathcal{T}} + (\text{cut})$. Then, by the cut elimination theorem, it is provable in $\text{JL}_{\mathcal{CS}}^{\mathcal{T}}$. \square

Inspection of all $\text{JL}_{\mathcal{CS}}^{\mathcal{T}}$ -tableau rules in Tables 3 and 2 shows that in a $\text{JL}_{\mathcal{CS}}^{\mathcal{T}}$ -tableau every expanded formula of a rule is a weak $\text{JL}_{\mathcal{CS}}$ -subformula of the root of the tableau.

Theorem 3.7 (Subformula property). *Every formula in a $\text{JL}_{\mathcal{CS}}^{\mathcal{T}}$ -tableau proof is a weak $\text{JL}_{\mathcal{CS}}$ -subformula of the root of the tableau.*

Note that the subformula property does not ensure decidability, because the number of $\text{JL}_{\mathcal{CS}}$ -subformulas of a formula is not necessarily finite. In fact, it is wrongly claimed in [8, page 172] that for a finite \mathcal{CS} , the set of all $\text{JL}_{\mathcal{CS}}$ -subformulas of a formula is always finite. For a counterexample, consider a formula $t : A$ and an empty \mathcal{CS} . The set of all JL_{\emptyset} -subformulas of $t : A$ includes $t : A, t : t : A, t : t : t : A, \dots$, which is obviously infinite.

4 Conclusion

We introduced two kinds of tableau proof systems for each justification logic JL , i.e. JL -tableaux of Section 3.1 and $\text{JL}^{\mathcal{T}}$ -tableaux of Section 3.2. We proved soundness and completeness theorems for both kinds of tableaux. While some JL -tableau rules are not analytic, we showed a kind of subformula property for $\text{JL}^{\mathcal{T}}$ -tableaux.

Acknowledgments

This research was in part supported by a grant from IPM. (No. 95030416)

References

- [1] S. Artemov, Operational modal logic, Technical Report MSI 95–29, Cornell University, 1995.
- [2] S. Artemov, Explicit provability and constructive semantics, *The Bulletin of Symbolic Logic*, 7(1), 1–36, 2001.
- [3] S. Artemov, The logic of justification, *The Review of Symbolic Logic*, 1(4), 477–513, 2008.
- [4] S. Artemov and M. Fitting, Justification logic, In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*, 2012.
- [5] M. D’Agostino, Are tableaux an improvement on truth-tables? Cut-free proofs and bivalence, *Journal of Logic, Language and Information*, 1, 235–252, 1992.
- [6] M. D’Agostino, Tableau methods for classical propositional logic, In *Handbook of Tableau Methods*, M. D’Agostino, D. Gabbay, R. Haehnle, and J. Posegga, eds, pp. 45–124. Kluwer, 1999.
- [7] M. D’Agostino and M. Mondadori, The taming of the cut. Classical refutations with analytic cut, *Journal of Logic and Computation*, 4, 285–319, 1994.
- [8] M. Finger, Analytic methods for the logic of proofs. *Journal of Logic and Computation*, 20(1), 167–188, 2010.

- [9] M. Fitting, *First-Order Logic and Automated Theorem Proving*, Springer, 1996, Second Edition.
- [10] M. Fitting, The logic of proofs, semantically, *Annals of Pure and Applied Logic*, 132(1), 1–25, 2005.
- [11] M. Fitting, Reasoning with justifications, In David Makinson, Jacek Malinowski, and Heinrich Wansing, editors, *Towards Mathematical Philosophy, Papers from the Studia Logica conference Trends in Logic IV*, volume 28 of *Trends in Logic*, chapter 6, pages 107–123. Springer, 2009.
- [12] R. Kuznets, *Complexity Issues in Justification Logic*, PhD thesis, City University of New York, May 2008.
- [13] R. Kuznets and T. Studer, Justifications, ontology, and conservativity, In Thomas Bolander, Torben Braner, Silvio Ghilardi, and Lawrence Moss, editors, *Advances in Modal Logic*, Volume 9, pages 437–458. College Publications, 2012.
- [14] A. Mkrtychev, Models for the logic of proofs, In S. I. Adian, A. Nerode (Eds.), *Logical Foundations of Computer Science*, Vol. 1234 of *Lecture Notes in Computer Science*, Springer, 1997, pages 266–275.
- [15] B. Renne, *Tableaux for the Logic of Proofs*, Technical Report TR-2004001, CUNY Ph.D. Program in Computer Science, March 2004.
- [16] B. Renne, Semantic cut-elimination for two explicit modal logics, In Janneke Huitink and Sophia Katrenko, editors, *Proceedings of the Eleventh ESSLLI Student Session, 18th European Summer School in Logic, Language and Information (ESSLLI'06)*, pages 148–158, 2006.
- [17] R. M. Smullyan, *First-Order Logic*, Dover Publications, Inc., New York, 1995.
- [18] A. Troelstra and H. Schwichtenberg, *Basic Proof Theory*, Cambridge University Press, Amsterdam, 1996.